HAL open science

Type-theoretical natural language semantics: on the system F for meaning assembly

Christian Retoré

▶ To cite this version:

Christian Retoré. Type-theoretical natural language semantics: on the system F for meaning assembly. TYPES 2013, Apr 2013, Toulouse, France. pp.64–65. hal-00799685

HAL Id: hal-00799685
https://hal.science/hal-00799685

Submitted on 12 Mar 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Type-theoretical natural language semantics: on the system F for meaning assembly

Christian Retoré[1]*

Équipe Melodi, IRIT , UPS, 118 route de Narbonne, 31062 Toulouse Cedex 9
(& LaBRI, Université de Bordeaux, 351 cours de la Libération, 33405 Talence cedex)

Roughly speaking, the compositional semantics analysis of natural language consists in mapping a sentence to a logical formula which depicts its meaning. To do so, a lexicon endow each word with a partial formula (a λ-term over the base type $\mathtt{t}$ for propositions and $\mathtt{e}$ for individuals), and the (binary) parse tree of the sentence specifies for each node which subtree is the function and which one is the argument. Hence a λ-term corresponding to the whole parse tree, and by reduction one obtains a λ-term of type $\mathtt{t}$ which corresponds to a formula of higher order logic. This classical process which uses Church's representation of formulae as simply typed λ-terms is the basis of the so-called Montague semantics, see e.g. [7, Chapter 3].

But it is more accurate to have many individual base types rather than just $\mathtt{e}$. This way, the application of a predicate to an argument only happens when it makes sense. For instance sentences like *"The chair barks."* or *"Their five is fast."* are easily ruled out when there are several types for individuals by saying that *"barks"* and *"fast"* respectively require arguments of type *"dog"* and *physical or animated object.* Nevertheless, such a type system needs to incorporate some flexibility. Indeed, in the context of a football match, the second sentence makes sense, because *"their five"* may be understood as a player.

Accounts of these meaning transfers received a lot of attention since the 80's and some formal accounts were proposed in particular by Asher [1]. We also proposed an account using the system F of Girard (1971) [3] and we studied the compositional properties of such a system in particular for quantifiers, plurals and generic elements, as well as the related lexical issues like meaning transfers, copredication, fictive motion,... see e.g. [2, 8]. Our system works as follows: the lexicon provides each word with a main λ-term, the "usual one" which specifies the argument structure of the word, by using refined types (e.g. "sleeps: $\lambda x^{ani}\underline{sleeps}(x)$" requires an animated subject). In addition, the lexicon may endow each word with a finite number of λ-terms (possibly none) that implement meaning transfers. For instance a *"book"* may be turned into a physical object ϕ or into an informational content I, by constants of respective types $book \to I$ and $book \to \phi$ (sometimes the λ-terms are more complex than simple constants). That way a sentence like *"This book is heavy but interesting."* can be properly analysed. Some meaning transfers are declared to be *rigid* in the lexicon: rigidity prohibits the use of other meaning transfers thus we can block *"* Liverpool defeated Chelsea and decided to build new docks."* because the meaning transfer from a town to a football club is declared in the lexicon as rigid.

In such a setting it is very convenient to quantify over types, for instance quantifiers $\forall, \exists$ may be given a type $\Lambda\alpha.(\alpha \to \mathtt{t}) \to \mathtt{t}$ It also allows a factorised treatment of conjunction: each time an object x of type ξ can be viewed both as an object of type α (via an optional term $f_0 : \xi \to \alpha$) to which the property P applies and as an object of type β (via an optional term $g_0 : \xi \to \beta$) to which the property Q applies, one can express that x enjoys $P \wedge Q$. This polymorphic *"and"* simply is: $\Lambda\alpha\Lambda\beta\lambda P^{\alpha\to\mathtt{t}}\lambda Q^{\beta\to\mathtt{t}}\Lambda\xi\lambda x^{\xi}\lambda f^{\xi\to\alpha}\lambda g^{\xi\to\beta}.\ (\wedge^{\mathtt{t}\to\mathtt{t}\to\mathtt{t}}\ (P\ (f\ x))(Q\ (g\ x)))$. To sum up the logical system, we also have two layers but they slightly vary from the ones of Montague

*Funded by ANR projects Loci and Polymnie.

semantics. Our *meta logic* (a.k.a. glue logic) is system F with many base types t, e_i (instead of simply typed λ-calculus with t, and e) Our *logic for semantic representations* is many-sorted higher-order logic (e_i instead of a single sort e). For representing quantification, we actually prefer to use Hilbert's ϵ and τ-terms constructed with two constants $\epsilon, \tau : \Lambda\alpha.\ (\alpha \to \mathsf{t}) \to \alpha$ and one for generic elements [8]. An important but rather easy property holds: if the constants define an n-order q-sorted logic, any (η-long) normal λ-term of type t does actually correspond to a formula of n-order q-sorted logic (possibly $n = \omega$).

We preferred system F to modern type theories (MTT) used by Luo [5] or to the categorical logic of Asher [1] because of its formal simplicity and absence of variants — as the terms are issued from the lexicon by means of syntactic rules, F terms with a problematic complexity are avoided. There are two properties of Luo's approach [5] that would be welcome: a proper notion of subtyping, mathematically safe and linguistically relevant, and predefined inductive types with specific reduction rules. Subtyping is most welcome in particular to represent ontological inclusion (a *"human being"* is an *"animal"*, thus predicates that apply to *"animals"* also apply to *"human beings"*). Coercive subtyping as developed by Luo and Soloviev[10] sounds promising for F (and other notions as well [4]). The key property of coercive subtyping is that there is at most one subtyping map between any two complex types, provided that there is at most one subtyping map between any two base types. Predefined types, inductive types would be most welcome in our setting, e.g. integers as in Gödel's system T and finite sets of α-objects. Of course, F can encode such types, but such encodings are far from natural. Reduction in such a setting is related to the work of Soloviev and Chemouil [9] The key point is to show that normalisation and confluence are preserved and that there is no constant-free and closed term of a false type. We shall also illustrate the linguistic relevance of these extensions, which are already included in Moot's semantical and semantical parser for French. [6]

References

[1] Nicholas Asher. *Lexical Meaning in context – a web of words.* Cambridge University press, 2011.

[2] Christian Bassac, Bruno Mery, and Christian Retoré. Towards a Type-Theoretical Account of Lexical Semantics. *Journal of Logic Language and Information*, 19(2):229–245, April 2010. `http://hal.inria.fr/inria-00408308/`.

[3] Jean-Yves Girard. *The blind spot – lectures on logic.* European Mathematical Society, 2011.

[4] Giuseppe Longo, Kathleen Milsted, and Sergei Soloviev. Coherence and transitivity of subtyping as entailment. *Journal of Logic and Computation*, 10(4):493–526, 2000.

[5] Zhaohui Luo. Contextual analysis of word meanings in type-theoretical semantics. In Sylvain Pogodalla and Jean-Philippe Prost, editors, *LACL*, volume 6736 of *LNCS*, pages 159–174. Springer, 2011.

[6] Richard Moot. Wide-coverage French syntax and semantics using Grail. In *Proceedings of Traitement Automatique des Langues Naturelles (TALN)*, Montreal, 2010.

[7] Richard Moot and Christian Retoré. *The logic of categorial grammars: a deductive account of natural language syntax and semantics*, volume 6850 of *LNCS*. Springer, 2012. `http://www.springer.com/computer/theoretical+computer+science/book/978-3-642-31554-1`.

[8] Christian Retoré. Variable types for meaning assembly: a logical syntax for generic noun phrases introduced by "most". *Recherches Linguistiques de Vincennes*, 41:83–102, 2012. `http://hal.archives-ouvertes.fr/hal-00677312`.

[9] Sergei Soloviev and David Chemouil. Some Algebraic Structures in Lambda-Calculus with Inductive Types. In Stefano Berardi, Mario Coppo, and Ferruccio Damiani, editors, *TYPES*, volume 3085 of *Lecture Notes in Computer Science*, pages 338–354. Springer, 2003.

[10] Sergei Soloviev and Zhaohui Luo. Coercion completion and conservativity in coercive subtyping. *Annals of Pure and Applied Logic*, 1-3(113):297–322, 2000.